A Comparison of a Relaxation Solver and a Preconditioned Conjugate Gradient Solver in Parallel Finite Element Groundwater Computations

by

Fred T. Tracy
Engineer Research and Development Center
Major Shared Resource Center

Abstract

The typical high-performance computing finite element (FE) groundwater computation requires the repeated solution of a system of simultaneous, linear equations with thousands to millions of unknowns. This is also typically where the dominant amount of computer time is spent, thus making the solver one of the most critical parts of the FE program. Some older legacy codes still have a relaxation solver as their primary solver. This paper first shows how this relaxation solver can be completely inadequate where point sources/sinks (such as wells) are needed to model the application. It then describes the implementation of a parallel conjugate gradient (CG) solver for the system of equations, Ax = b, where A is symmetric and positive definite, using an incomplete lower-upper (ILU) factorization preconditioner. Finally, correct results and parallel performance using the CG solver are shown for a practical groundwater problem.

Introduction

The typical high-performance computing finite element (FE) groundwater computation requires the repeated solution of a system of simultaneous, linear equations with thousands to millions of unknowns. This is also typically where the dominant amount of computer time is spent, thus making the solver one of the most critical parts of the FE program. Some older legacy codes still have a relaxation solver as their primary solver. Although the state-of-the-art solvers converge much faster, the relaxation solver is not replaced because it is simple and perceived to be a good, despite being old, workhorse. Also, a parallel version of the relaxation solver where one does domain decomposition with Dirichlet boundary conditions on the boundary degrades some as one increases the number of processing elements (PEs). Yet, it is not so bad that one would necessarily get the inertia to replace it for that reason alone. However, attaining remarkably poor results is a good reason to provide another option. This paper will first show how poorly this solver does for some very important problems in groundwater modeling. It will then describe the implementation of a parallel conjugate gradient (CG) solver using an incomplete lower-upper (ILU) factorization preconditioner. Finally, performance results will be given for a practical groundwater problem.

Governing Equations

Steady-state flow of groundwater is the primary illustrative problem in this study, and it is represented by the equation

$$\nabla \cdot (\mathbf{k} \cdot \nabla\phi) = 0 \tag{1}$$

where

$\mathbf{k}$ = hydraulic conductivity tensor

ϕ = potential or total head

The standard Galerkin finite element formulation produces a set of symmetric, positive-definite system of equations represented by

$$[K]\{\phi\} = \{Q\} \tag{2}$$

where

[K] = stiffness matrix

{ϕ} = vector of total head (L) values at the nodes

{Q} = vector of external flow (L^3/T) values at the nodes where head is not specified

{Q} is obtained from the boundary conditions. {Q} contains the total head value at the respective nodes where it is specified, and [K] is modified so it remains symmetric near these specified total head nodes.

"Worst Case" Sample Problem with Comparative Results

Figure 1 shows the top view of an FE mesh for the program of a regional groundwater problem, and Figure 2 shows an isometric view of a portion of the mesh for the program FEMWATER (Lin et al. 1997) in the Groundwater Modeling System (Groundwater Modeling Team 2001). The different colors represent different soil types. This is a relatively small mesh of 18,752 nodes and 31,787 elements. The red diamonds represent where total head is specified, equivalent to a line source or sink such as a fully penetrating well with known drawdown. For ease of comparison, a total head of 200 was placed at the lower, left nodes, and a total head of 100 was placed at the upper, right nodes. The initial total head was set to 200. The rest of the boundary is impervious to flow. The actual flow region has wells in the interior with many other boundary conditions and rainfall input. These simpler boundary conditions were created so the point could be better illustrated.

Figure 3 shows a total head color contour plot of the plan view from the ILU CG solver where it converged in 341 iterations at a tolerance of 0.00001 in a few seconds on one PE on the Cray T3E. Figure 4 shows the isolevel line contours for the isometric view of this same run. These results are correct.

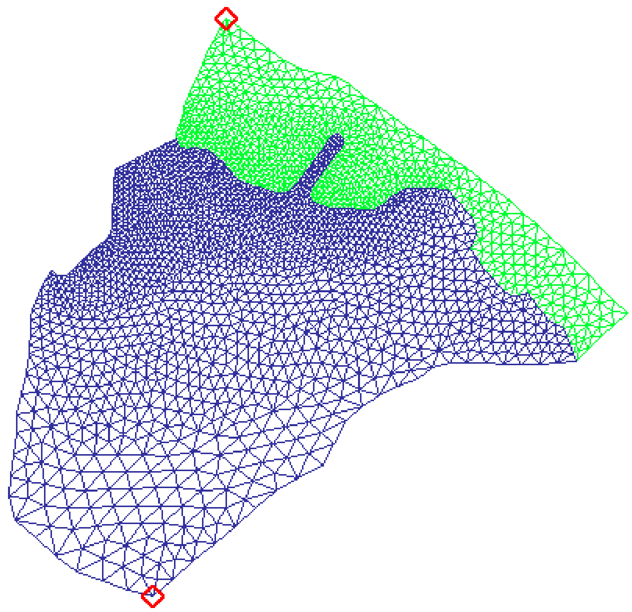

Figure 1. Regional Groundwater Problem FE Mesh

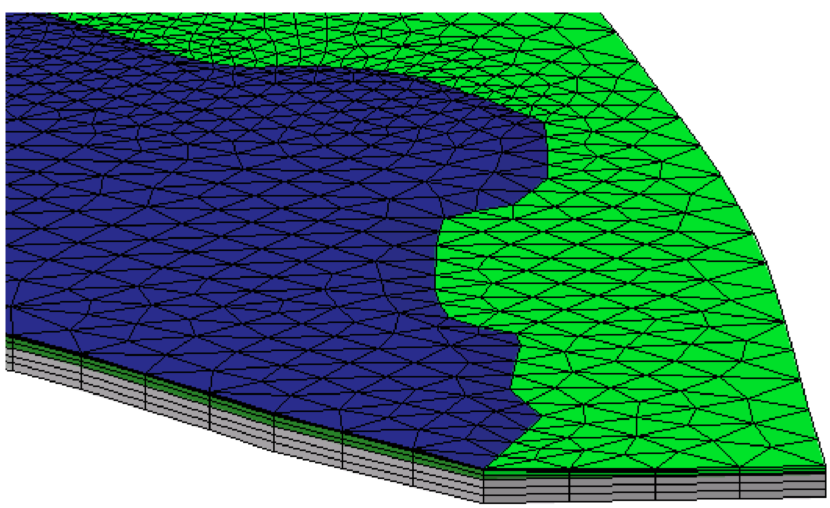

Figure 2. Isometric View of a Portion of the Mesh

Now when the relaxation solver was used, the results were amazingly and deceptively poor because of the slow convergence. Figure 5 shows the results of the relaxation solver after

5,000 iterations, and Figure 6 shows the results of the relaxation solver after 10,000 iterations. One iteration consists of a forward sweep from the first node to the last node followed by a backward sweep from the last node to the first node, similar to the alternating direction implicit (ADI) algorithm for structured grids. These plots should look almost identical to Figure 3, but they are both significantly different. Table 1 shows the maximum absolute error and the pressure head (total head minus the elevation) for a typical node (see Node 950 in Figure 7). The deception is that when the absolute errors are rather small, the errors in the results are still remarkably large. This is sometimes lost when looking at the results of a complex problem. The large number of iterations indicates the extremely slow convergence. The 50,000-iteration run took 3186.1 sec on one PE of the T3E, and it is still has a percent difference from the CG solution of 2.4 percent.

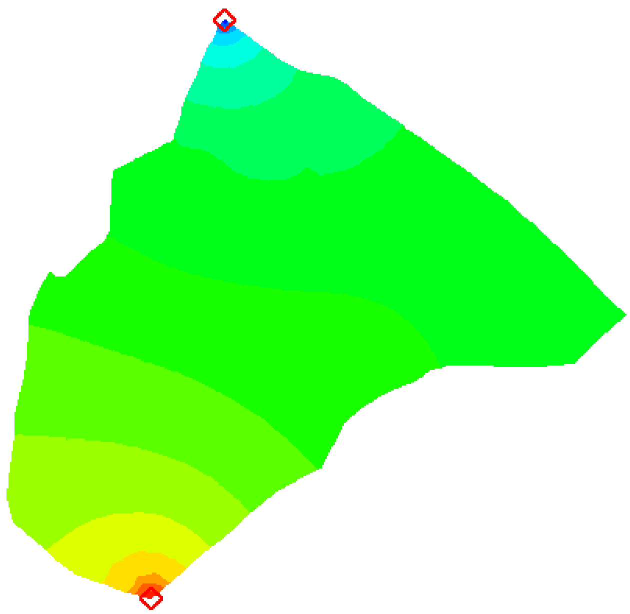

Figure 3. Color Contour of Total Head for the ILU CG Solver

Figure 4. Isolevel Line Contours for the Isometric View

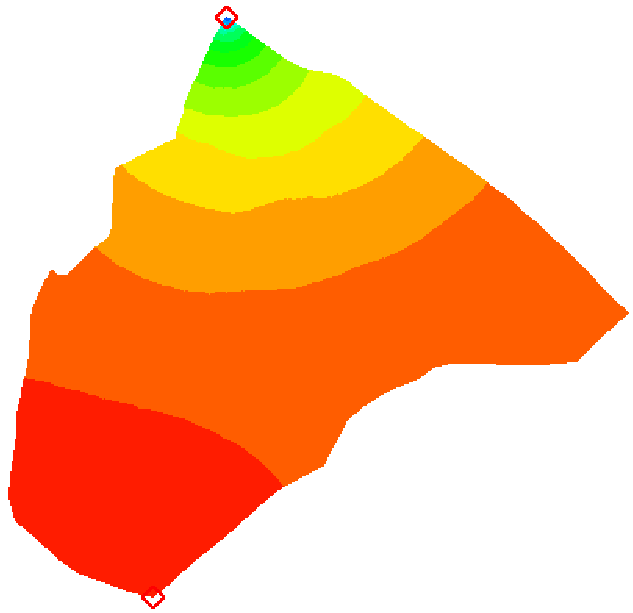

Figure 5. Relaxation Solver after 5,000 Iterations

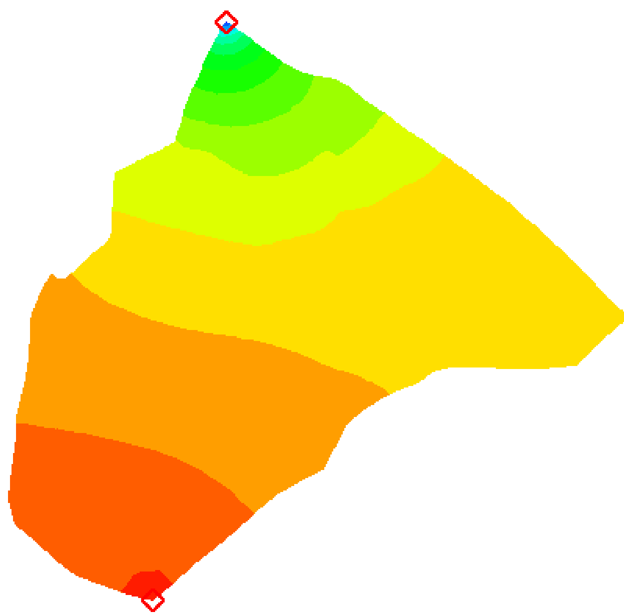

Figure 6. Relaxation Solver after 10,000 Iterations

A well is also often modeled by specifying the quantity of flow. Table 2 shows the results for the two solvers when flow is specified where total head = 100 was used in the previous example. After 50,000 iterations, the pressure head was still off by 6.4 percent.

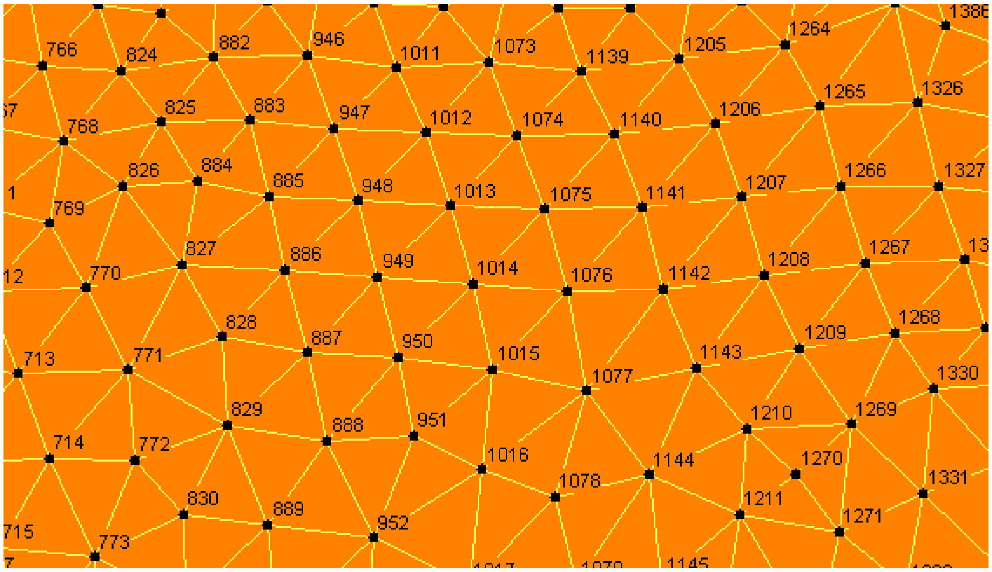

Figure 7. Node 950 at the Top and Middle of the Mesh

Other problems will not be this severe, but knowing to what extent this effect will be exhibited is challenging, so another solver is essential.

Parallel ILU CG Solver

A parallel version of FEMWATER for both flow and transport (Tracy 2000) has been developed. In both the serial and parallel version, the ILU CG solver (Dongara, Sorensen, and van der Vorst 1998) mentioned previously has been implemented. The preconditioner consists of approximating the original [K] given in Equation 2 with

$$[\tilde{K}] = [\varpi L + D][D]^{-1}[D + \varpi U], \quad 0 \le \varpi \le 1 \qquad (3)$$

where

[L] = lower part of [K]

[D] = diagonal part of [K]

[U] = upper part of [K]

ϖ = relaxation-type factor

A CG iteration involves first the solution for {y} of the tridiagonal system

$$[D + \varpi U]\{y\} = \{r\} = \{Q\} - [K]\{\phi\} \qquad (4)$$

where

$\{r\}$ = residual

This step is followed by the solution for $\{z\}$ of the tridiagonal system

$$[\varpi L + D]\{z\} = [D]\{y\} \tag{5}$$

SPECIFIED HEAD			
Iterations	**Maximum Absolute Error**	**Pressure Head at Node 950 (CG Value = 144.4)**	**Percent Difference**
5,000	0.002347	181.8	22.9
10,000	0.001818	173.1	18.1
15,000	0.001310	166.4	14.2
20,000	0.000960	161.3	11.1
25,000	0.000768	157.4	8.6
30,000	0.000590	154.4	6.7
35,000	0.000453	152.1	5.2
40,000	0.000347	150.0	3.8
45,000	0.000267	148.9	3.1
50,000	0.000205	147.9	2.4

Table 1. Absolute Errors and Percentage Differences for Specified Head

SPECIFIED FLOW			
Iterations	**Maximum Absolute Error**	**Pressure Head at Node 950 (CG Value = 155.5)**	**Percent Difference**
5,000	0.000983	188.8	19.3
10,000	0.000836	184.7	17.2
15,000	0.000726	181.1	15.2
20,000	0.000629	178.0	13.5
25,000	0.000546	174.8	11.7
30,000	0.000482	172.8	10.5
35,000	0.000429	170.7	9.3
40,000	0.000369	168.8	8.2
45,000	0.000331	167.2	7.3
50,000	0.000290	165.8	6.4

Table 2. Absolute Errors and Percentage Differences for Specified Flow

Equations 4 and 5 do not parallelize very well. To see what is done, one should consider Figure 8 showing a partitioning of one of the tridiagonal systems between two PEs with the asterisks representing nonzero terms. What is done is to solve the tridiagonal systems inside each PE only with terms crossing into another PE set to zero. In the example in Figure 8, the lower, left-hand and upper, right-hand sections are all set to zero. This has the effect of the preconditioner being somewhere between

$$\left[\tilde{K}\right] = [D] \tag{6}$$

and the full ILU preconditioner represented by Equation 3.

$$\begin{array}{l} \\ \\ \text{PE 0} \\ \\ \\ \\ \text{PE 1} \\ \\ \\ \end{array} \left[\begin{array}{ccccc|cccc} * & * & 0 & * & * & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & 0 & 0 & 0 \\ 0 & * & * & 0 & * & * & 0 & 0 & 0 \\ * & * & 0 & * & * & 0 & * & * & 0 \\ * & * & * & * & * & * & * & * & * \\ \hline 0 & * & * & 0 & * & * & 0 & * & * \\ 0 & 0 & 0 & * & * & 0 & * & * & 0 \\ 0 & 0 & 0 & * & * & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * & 0 & * & * \end{array}\right] \{y\} = \{r\}$$

Figure 8. Partitioning for Two PEs

Parallel Performance

Parallel FEMWATER was developed under the Common High Performance Computing Software Support Initiative, and its beta test problem was the remediation of a large military site (Tracy et al. 1999) where the plan view of the mesh is shown in Figure 9. Various numbers of layers were used in the full three-dimensional flow simulation. For this study, this mesh consists of 102,996 nodes and 187,902 elements. The test is to run 20 nonlinear iterations with 500 nonlinear iterations, as this problem is unsaturated flow. The problem could not fit on less than eight PEs on the T3E. Table 1 shows the parallel speedup results for the T3E when keeping the same problem size and increasing the number of PEs. Table 2 shows the scaled speedup (actual wall clock time divided by the ideal wall clock time times 100 percent) results for the T3E where the number of PEs and the problem size (number of elements) are both doubled. Ideally, the running times would not increase as the number of PEs is increased. However, the communication typically increases with the number of PEs. Sometimes the running time will actually go down (scaled speedup greater than 100 percent) as in this example when going from eight to 16 PEs because of cache memory differences.

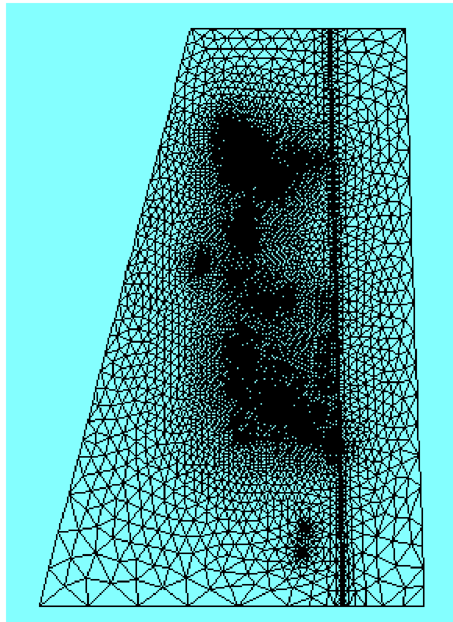

Figure 9. Plan View of the Mesh

Conclusion

The ILU CG solver does a good job at solving various groundwater problems, as the relaxation solver has a difficult time converging for some problems. The parallel and scaled speedup results are excellent for the moderate number of PEs tested.

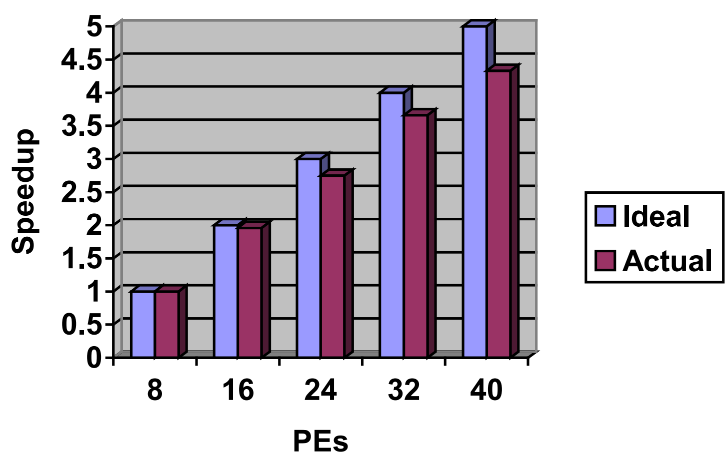

Table 1. Parallel Speedup

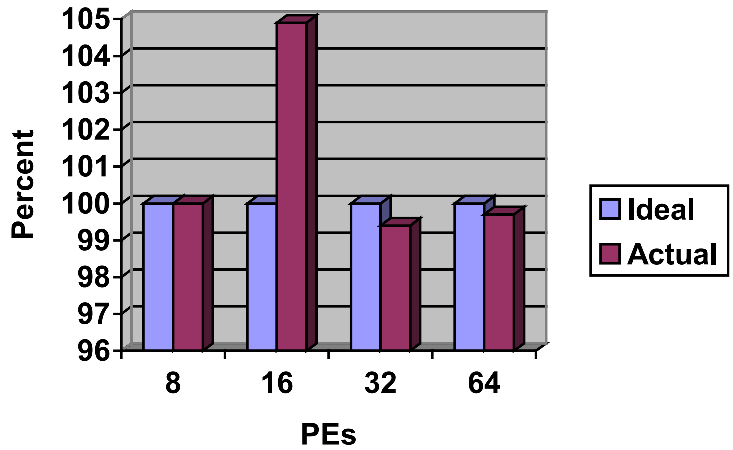

Table 2. Scaled Speedup

Acknowledgment

This work was supported in part by a grant of computer time from the DoD High Performance Computing Modernization Program at the U.S. Army Engineer Research and Development Center Major Shared Resource Center, Vicksburg, MS.

References

1. Lin, H. J., Richards, D. R., Talbot, C. A., Yeh, G. T., Cheng, J. R., Cheng, H. P., and Jones, N. L. (1997). "FEMWATER: A three-dimensional finite element computer model for simulating density-dependent flow and transport in variably saturated media," Technical Report CHL-97-12, U.S. Army Engineer Research and Development Center (ERDC), Vicksburg, MS.

2. Groundwater Modeling Team. (2001). *GMS*, http://chl.wes.army.mil/software/gms/, U.S. Army Engineer Research and Development Center (ERDC), Vicksburg, MS.

3. Tracy, F. T. (2000). *Parallelization of finite element method flow and transport groundwater computations*. Southern Conference on Computing, University of Southern Mississippi, Hattiesburg, MS.

4. Dongara, J. J., Sorensen, D. C., and van der Vorst, H. A. (1998). *Numerical linear algebra for high-performance computers*. SIAM, Philadelphia, 203.

5. Tracy, F. T., Talbot, C. A., Holland, J. P., Turnbull, S. J., McGehee, T. L., and Donnell, B. P. (1999). *The application of the parallelized groundwater model FEMWATER to a deep mine project and the remediation of a large military site*. DoD HPC Users Group Conference Proceedings, Monterey, CA.